CEC Continuing Education Center

DesignNews

Secure MCUs and RTOSs

DAY 1 : Threat Model Security Analysis (TMSA)

Sponsored by

Digi-Key ELECTRONICS

BENINGO EMBEDDED GROUP

informa markets

Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Group Chat’ by maximizing the chat widget in your dock.

THE SPEAKER

Jacob Beningo

Visit ‘Lecturer Profile’

Beningo Embedded Group - President

Focus: Embedded Software Consulting

An independent consultant who specializes in the design of real-time, microcontroller based embedded software.
He has published two books:

- Reusable Firmware Development
- MicroPython Projects
- Embedded Software Design

Writes a weekly blog for DesignNews.com focused on embedded system design techniques and challenges.

Visit www.beningo.com to learn more …

Visit ‘Lecturer Profile’ in your console for more details.

Course Sessions

- **Threat Model Security Analysis (TMSA)**
- Secure Microcontroller Solutions
- Arm TrustZone
- Secure Boot and Firmware Updates
- Secure RTOSes

1 Security Introduction

Security Introduction – Security isn't optional anymore

Billions of IoT devices

Data integrity, security and privacy

Potential losses from Hacks and breaches

Government Regulation

Security Introduction – Securing the IoT

Securing a physical switch is simple, lock-it out or restrict access to it. Securing a Smart IoT Switch is more complicated, suddenly have:

- Light and Proximity sensors
- Logic implemented on a microcontroller
- Wireless connectivity
- Network communication
- Cloud based application and services
- The need to secure the system and its data

Security Introduction – Securing the IoT

Securing a system can seem complicated and overwhelming!
Developers suddenly have to know:

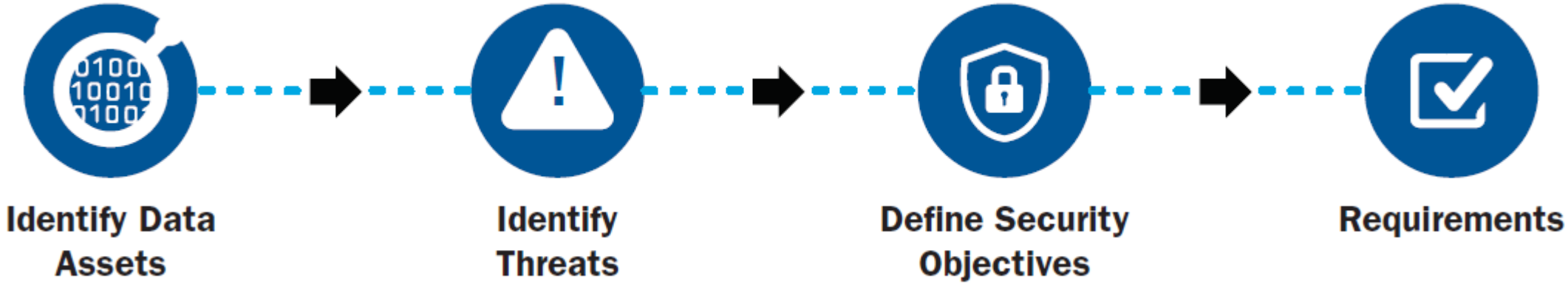
- Attack categories
- Cryptography
  - Hashes
  - Signatures
  - Encryption
- Secure boot
- Secure firmware updates

How can developers judge how much security they need?

How important is security in your embedded systems?
- Not applicable, I don't make connected devices
- Basic security is needed
- Definitely important
- Critical to our business and customer
- Other

2 Threat-Based Analysis

Threat-Based Security Analysis



Threat-Based Security Analysis

Confidentiality - the state of keeping or being kept secret or private.

- requires that only authorized people can read the data asset.
- it is kept secret or private
- Data assets that require confidentiality include:
  - Passwords
  - Personal data generated by the IoT device
    - Heart rate
    - Location data

Threat-Based Security Analysis

Integrity – the state of being whole and undivided.

- requires that a data asset remains unchanged through its use or transferal.
- Data assets that require integrity include:
  - Boot firmware (ensures that the MCU initializes to a known initial state)
  - Device configuration
  - Credentials
  - Firmware
  - etc

Threat-Based Security Analysis

Authenticity – the quality of being authentic (undisputed origin; not a copy, genuine)

- Requires that only a trusted actor has established the current state of a data asset
- Example data assets requiring authenticity include:
  - Firmware images

Combining integrity and authenticity establishes trust!

- Digital signature can be used to evaluate and integrity of new firmware
- Digital signature can be used to evaluate and integrity of existing firmware

How familiar are you with CIA?
- Beginner
- Intermediate
- Expert
- Other

3 An Example Analysis

Threat-Based Security Analysis – A Networked Camera

- Used to stream live video to a remote location
- May be used to periodically capture still images or be activated by detected motion
- Video stream is transmitted in a compressed form
- May be used for:
  - Personal use (baby monitors, doorbells, security, etc
  - Enterprise general use (security, event detection)
  - Enterprise high security use (protect high value assets)

An Example – Step #1 – Identifying the Data

Data assets that exist in nearly all IoT devices include:

- The firmware
- Unique ID
- Passwords (flash, users, etc)
- Encryption keys (to control device, secure communication, etc)

Device specific data assets might include:

- Image data
- Sensor data
- Control data

An Example – Step #1 – Identifying the Data

Data assets that exist in a Networked Camera:

Data Asset	Confidentiality	Integrity	Authentication
Camera Id		✓	
Firmware	✓	✓	✓
Firmware Credentials		✓	
Credentials	✓	✓	
Logs		✓	
Images	✓	✓	
Configuration	✓	✓	

An Example – Step #2 – Identifying the Threats

Threat	Targeted Data Asset	Confidentiality	Integrity	Authentication
Impersonation	Credentials	✓	✓	
Man in the Middle	Credentials			
	Images			
	Configuration	✓	✓	
	Confidentiality	✓	✓	✓
Firmware Abuse	Firmware	✓	✓	✓
Tamper	Camera ID		✓	
	Firmware	✓	✓	✓
	Credentials	✓	✓	✓
	Logs		✓	
	Images	✓	✓	
	Configuration	✓	✓	

An Example – Step #3 – Identifying Security Objectives

Access Control - The IoT device authenticates all actors (human or machine) attempting to access data assets. Prevents unauthorized access to data assets. Counters spoofing and malware threats where the attacker modifies firmware or installs an outdated flawed version.

Secure Storage - The IoT device maintains confidentiality (as required) and integrity of data assets. Counters tamper threats.

Firmware Authenticity - The IoT device verifies firmware authenticity prior to boot and prior to upgrade. Counters malware threats.

Communication - The IoT device authenticates remote servers and provides confidentiality (as required) and maintains integrity of
exchanged data. Counters Man in the Middle (MitM) threats.

Secure State - Ensures that the device maintains a secure state even in case of failure of verification of firmware integrity and
authenticity. Counters malware and tamper threats.

An Example – Step #4 – Defining Requirements

Security Objective	Countered Threats	Targeted Data Assets	Security Properties[2]	Design	Mfg	Inventory	End Use	Term
Access Control[1]	Spoofing Malware	Configuration T. Firmware	C I, A	N/A Dig Sign	N/A Dig Sign	N/A N/A	Encryption Dig Sign	Dead[4] Dead[4]
Secure Storage[1]	Tamper	HW ID T. Firmware User Data Configuration Keys	I I, A C, I C C, I	N/A Dig Sign N/A N/A N/A	eFuse Dig Sign N/A N/A SEF[3]	eFuse Dig Sign N/A N/A SEF[3]	eFuse Dig Sign Encryption Encryption SEF[3]	eFuse Dead[4] Dead[4] Dead[4] Dead[4]
Firmware Auth	Malware	T. Firmware	I, A	Dig Sign	Dig Sign	Dig Sign	Dig Sign	Dead[4]
Comm[1]	MitM	User Data Keys	C, I C, I	N/A N/A	N/A SEF[3]	N/A SEF[3]	Encryption SEF[3]	Dead[4] Dead[4]
Secure State	Malware Tamper	T. Firmware HW ID T. Firmware User Data Configuration Keys	I I, A I, A C, I C C, I	Dig Sign N/A Dig Sign N/A N/A N/A	Dig Sign eFuse Dig Sign N/A N/A SEF[3]	Dig Sign eFuse Dig Sign Encryption Encryption SEF[3]	Dig Sign eFuse Dig Sign Encryption Encryption SEF[3]	Dead[4] eFuse Dead[4] Dead[4] Dead[4] Dead[4]

What step do you find you pay the least attention to?
- Step #1
- Step #2
- Step #3
- Step #4
- Other

4 Going Further

Security and RTOS Resources

- Jacob's RTOS Blogs
- Jacob's RTOS courses
- Jacob's Security Blogs
- TrustZone for Cortex-M

- Embedded Bytes Newsletter
  - http://bit.ly/1BAHYXm

BENINGO EMBEDDED GROUP

www.beningo.com

CEC Continuing Education Center

DesignNews

Thank You

Sponsored by

Digi-Key ELECTRONICS

BENINGO EMBEDDED GROUP

informa markets